
Markov Economies and Stochastic Dynamical Systems

*Riverrun, past Eve and Adam's, from swerve of shore to bend
of bay, brings us to a commodius vicus...*

James Joyce

10.1 Introduction

Markov chains, and more generally stochastic processes, are immensely powerful tools for characterizing aspects of dynamical systems that cannot be captured using systems of ordinary differential equations. While the topic is far too broad to deal with in any detail in this book, I have found that some introduction to such models, which may be called *stochastic dynamical systems* is important if students are to develop a mature sense of how one models probabilistic social systems. Students who like what they see here can proceed to a more leisurely study in Karlin and Taylor (1975,1981) which gives many examples from economics, finance, and biology.

This chapter samples the literature by presenting two examples. The first is a model of my own that generalizes a famous paper by Kiyotaki and Wright (1989) concerning the emergence of money in a barter economy. This model is an interesting hybrid of Markov chain theory and game theory, since we use the ergodic theorem for finite Markov chains (§16.19) to define the payoffs to the game under a variety of conditions, and investigate the properties of Nash equilibria subject to these payoffs without specifying out-of-equilibrium behavior of the model. The Nash equilibria are thus ‘rational expectations equilibria,’ in which the expectations are stationary distributions of the Markov chain. To show that the model has plausible disequilibrium behavior, we present an artificial life simulation of the theoretical results.

VISIT...

LANZAROTE
Caliente.COM

Our second example is a Markov model of adaptive learning that illustrates the concept of stochastic stability, as developed in Young (1998). After developing some of the theoretical results, we again provide an artificial life simulation the model.

10.2 The Emergence of a Money in a Markov Economy

The neoclassical general equilibrium model is limited by its failure to model strategic interactions among agents (§3.17, §6.14.1, and §6.14.3). Typical of this weakness is its difficulty in modeling money. General equilibrium theory has generally treated money as an asset that is held because of its low risk and high liquidity. But money is dominated by demand deposits that have the same risk and liquidity characteristics as money, but unlike money are also interest-bearing (Chang, Hamberg, and Hirata 1983). In fact *money lowers transactions costs in a world in which it is expensive to write and enforce two-sided exchanges of goods*. Using game theory, we can develop a stochastic model of production and exchange that illustrates the transactions demand for money and that shows how money might emerge through a dynamic evolutionary process (Gintis 1997).

Consider an game with a large but finite number of players, each of whom lives a finite but uncertain number of periods. In each period players are randomly paired and engage in bilateral exchange. Each player produces some goods, consume others, and holds an inventory of untraded goods. We show that under plausible conditions a Nash equilibrium exists in which a good is a medium of exchange, accepted in trade by players who value it only for its ability to facilitate future trades with other agents.¹

Each player has a strategy specifying which trades are acceptable from a given pre-trade inventory, and what the player consumes and produces from a given post-trade inventory. Players meet randomly in each period with their accumulated inventories, they trade, and they then produce and consume, given their post-trade inventories. We assume each player has a constant probability ρ of living to the next period, and a player who dies is immediately replaced by an identical player with an empty inventory, who inherits the strategy of its predecessor.

Let G be the (finite) set of goods in the economy, each available in an integral number of units, and let K be a finite set of players. Each player

¹This model is inspired by Kiyotaki and Wright 1989. See also Kiyotaki and Wright (1991,1993), and Aiyagari and Wallace (1991,1992).

$k \in K$ consumes goods $C_k \subset G$ and produces goods $P_k \subset G$. Players do not consume the goods they produce—whence the need for trade. Each player $k \in K$ has an increasing utility function u_k over C_k . Each good $g \in P_k$ has unit production cost $\kappa_g > 0$, measured in utility units, and we assume every player k has some good $g_k \in P_k$ such that $\kappa_{g_k} < u(g)$ for some $g \in C_k$, so gains from trade are possible. If p is a bundle of goods consisting of n_g units of each good $g \in G$, we write $p = \sum_{g \in G} n_g g$. The cost of producing $p = \sum_{g \in P_k} n_g g$ is then

$$\kappa(p) = \sum_{g \in P_k} n_g \kappa_g.$$

In addition, each player $k \in K$ incurs a storage cost $\alpha_g > 0$ in each trading period for each unit of a good g held in inventory. Thus the cost of holding inventory $\mathbf{i} = \sum_{g \in G} n_g g$ is

$$\alpha(\mathbf{i}) = \sum_{g \in G} n_g \alpha_g,$$

We assume inventories can hold a maximum of n_{max} items. The utility accruing to player $k \in K$ in one round is

$$v_k = u_k(c_k) - \kappa(\gamma_k) - \alpha(\mathbf{i}_k), \quad (10.1)$$

where c_k is the set of goods consumed, γ_k is the set of goods produced, and $\mathbf{i}_k$ is the inventory k takes into the next trading period.

Players are randomly paired for trade in each period. A pure strategy $s_k \in S_k$ for player $k \in K$ consists of the following for each inventory $\mathbf{i}$: a trading function $A_k(\mathbf{i})$ such that if inventory $\mathbf{j} \in A_k(\mathbf{i})$, then k is willing to trade so as to move from inventory $\mathbf{i}$ to inventory $\mathbf{j}$; a consumption function $c_k(\mathbf{j})$ that indicate what k consumes (if anything) from post-trade inventory $\mathbf{j}$; and a production function $\gamma_k(\mathbf{j})$ that indicates what goods k produces to add to post-trade inventory $\mathbf{j}$. Notice that there are only a finite number of pure strategies for a player, since there are only a finite number of inventories.

If players k and k' using strategies $s_k \in S_k$ and $s_{k'} \in S_{k'}$ meet and they hold inventories $\mathbf{i}_k$ and $\mathbf{i}_{k'}$ respectively, their trading functions A_k and $A_{k'}$ defining a (possibly empty) set of mutually acceptable trades $G(s_k, \mathbf{i}_k, s_{k'}, \mathbf{i}_{k'})$. If this set is nonempty, a trade is chosen randomly from among its Pareto-efficient members.

If $s = \{s_k | k \in K\}$ is a pure strategy for the game, and if we start off each agent with an empty inventory, the model will generate a time path

of consumption, production, exchange, and inventories for each player in each future time period, depending on the random assignment of players to trade and the choice of trades from the sets of mutually acceptable trades. This can be a very large game with a huge number of strategies and lots of moves by Nature.

However there is an easy way to make the model tractable: we ask what happens in long-run equilibrium, when the pattern of consumption, production, and trade have become in some appropriate sense *time independent*. In such a situation, we can assume each player knows the probability distribution over player types and inventories, and chooses production, consumption, and trading strategies to maximize long-run expected utility, where expectations are taken relative to this probability distribution.

But how to we obtain such a probability distribution? We define the *state* of the game at a point in time to be the inventory holdings $\mathbf{I} = \{\mathbf{i}_k | k \in K\}$ of each of the players. Then the game strategy s generates a finite probability transition matrix P^s such that $P^s(\mathbf{I}, \mathbf{I}')$ is the probability of moving from $\mathbf{I}$ to $\mathbf{I}'$. The random variable $\mathbf{I}$ determined by P^s is then a *finite Markov chain*, and by the *ergodic theorem for finite Markov chains* (see Theorem 16.1), there is a stationary distribution $\mathcal{P}^s$ over inventories that is constant from the current to the next period. Specifically, for any inventory $\mathbf{I}$ the probability $P^+(\mathbf{I})$ that $\mathbf{I}$ occurs in the next period, which is given by

$$P^+(\mathbf{I}) = \sum_{\mathbf{I}'} P^s(\mathbf{I}') P^s(\mathbf{I}', \mathbf{I}),$$

satisfies $P^+(\mathbf{I}) = P^s(\mathbf{I})$.² The ergodic theorem also states that after the game is run for a sufficiently long time, the expected distribution of inventories will be equal to the stationary distribution P^s , so it is reasonable for a player to use P^s to plan an optimal production, trading, and consumption strategy.

These definitions extend directly to the case where each player uses a mixed strategy σ_k , giving rise to the game-level mixed strategy $\sigma = \{\sigma_k | k \in K\}$, and the ergodic distribution P^σ over inventories. We then define the payoff to player $k \in K$, given σ , to be the player's expected return from using strategy σ_k where the probability distribution

²Since by construction every state in the Markov chain has a nonzero probability of transition to the zero state, the set of states attainable from the zero state is an irreducible aperiodic persistent set, so the ergodic theorem applies.

of moving from inventory $\mathbf{i}_k$ to $\mathbf{i}'_k$ is given by the marginal probability distribution induced by P^σ on k 's inventories.

We have in principle defined a normal form game ('in principle' because the relevant equations have not been written out in detail; the reader is encouraged to do so). Clearly there is a *no trade Nash equilibrium* in which no player produces, consumes, or trades anything. However there are lots of other Nash equilibria, which is suggested by the following theorem, which follows trivially from Nash's theorem that finite games always have at least one Nash equilibrium (§4.3):

Theorem 10.1. *Suppose each player $k \in K$ is constrained to a nonempty pure strategy set $U_k \subseteq S_k$. Then there exists a Nash equilibrium in mixed strategies based on the pure strategies in $\{U_k \mid k \in K\}$.*

For instance we have

Theorem 10.2. *Suppose each player produces exactly one good, and for any two goods g and h there is a producer of g that consumes h . If the utility of consumption is sufficiently great for each good, there is a Nash equilibrium in which all goods are traded with positive probability.*

We leave the proof of this, which is straightforward, to the reader—or see Gintis (1997). Hint: constrain the pure strategies of the players by saying that each player must hold a unit of some good in its inventory in each period, and a player must offer to trade its production good g_k unit for unit against any consumption good $c_k \in C_k$. By Theorem 10.1 there exists a Nash equilibrium for the resulting 'constrained' game. Then show that when the constraints are relaxed, the players have no incentive to change their strategies.

What about money? We say a good $m \in G$ is a *universal medium of exchange*, or simply is *money*, if m is accepted in trade by all agents against all other goods $g \neq m$. We say a good $m \in G$ has *low storage costs* if for each $k \in K$, $s_m \leq s_g$ for all $g \in G$. We then have

Theorem 10.3. *Suppose each player $k \in K$ produces exactly one good g_k , and let $m \in G$ have low storage costs. Suppose that for each $g \in G$ there is an player who produces m and consumes g . Then if the utility of consumption is sufficiently high, there is a Nash equilibrium of the economy in which m is a universal medium of exchange.*

We again leave the proof to the reader. Hint: suppose $m \in G$ satisfies the conditions of the theorem. Add three constraints to player's pure

trade strategies. First, constrain each player to hold a unit of some good in its inventory in each period. Second, for each player $k \in K$, admit only those trade preferences A_k for which k 's post-trade inventory is no smaller than k 's pre-trade inventory. Third, require all players to choose strategies in which one unit of m is accepted in exchange for each good $g \neq m$ that they currently hold in their inventories. Then show that when the constraints are relaxed, players have no incentive to change their strategies.

What about *fiat money*, which is a money that is neither produced nor consumed by players (presumably it is produced by some outside agent, such as the government)? We assume there is a *fiat* good that may be traded, and upon the death of an player, is passed on to the player's successor. We say the good is *fiat money* if it is accepted in trade by all agents.

Theorem 10.4. *Suppose there is a fiat good m , each player produces exactly one good, each good except m is produced by at least one player, and for any two goods $g, h \neq m$ there is a producer of g that consumes h . If the amount of m is sufficiently small, the storage cost of m is sufficiently low, and the utility of consumption is sufficiently high, there is an equilibrium in which each good $g \neq m$ trades for m with positive probability in each period; i.e., m is fiat money.*

We leave the proof to the reader (Hint: Add three constraints to player's pure trade strategies. First, constrain each player to hold a unit of some good in its inventory in each period. Second, constrain players to offer only one unit of a good in trade. Third, require all agents to accept a unit of m in exchange for some good $g \neq m$ currently held in inventory. Show that in the resulting equilibrium, no agent gains by deviating from its constrained strategy).

10.2.1 Simulating A Monetary Economy

It is nice to show that trading systems with money exist, but it is quite another to show that among the myriad of Nash equilibria of such systems, equilibria with money have some privileged position. The strongest such statement would be that under a suitable dynamic there is convergence to a Nash equilibrium with money for a reasonable range of initial conditions. The emergence of money need not be a high probability event, since trading systems with money will outperform barter and nontrade systems, and hence will have a tendency to expand at their

expense. But local stability over a range of initial conditions would be highly desirable.

Unfortunately we lack the analytical machinery to prove local stability in the context of the above model, but we can show through simulations that money always emerges when the appropriate conditions are present. To show this, we construct an economy with five goods $g_1, \dots, g_5$, ordered from highest to lowest in storage costs. We assume that, except in the case of a *fiat* good, for each pair of goods there is a type of player in the economy producing the first good and consuming the second. There are thus twenty player types, since we assume that for each good g , the utility of consumption u_g , the cost of production κ_g , and the storage cost s_g are the same for all players who consume, produce, and store g , respectively.

For simplicity we assume that an player's maximum inventory is one unit, and we constrain each player to consume when it has acquired its consumption good, and to produce a unit of its production good when its inventory is empty. Dropping these constraints merely lengthens the time to convergence, since players who act otherwise are eventually driven into extinction. This reduces a player's strategy to its trading function—the specification of which goods the player will accept in exchange for its inventory. We call such a trading strategy the player's *genotype*, and we assign genotypes randomly to the initial population. We represent the genotype by a 5×5 matrix $\gamma = (\gamma_{ij})$, where $\gamma_{ij} = 1$ if the player accepts good g_j in trade for g_i and $\gamma_{ij} = 0$ if the player refuses such a trade.

With five goods there are $2^{25} = 33,554,432$ distinct trading strategies. We can reduce this number to $2^{16} = 65,536$ by eliminating dominated strategies (for instance strategies that involve not accepting a consumption good in trade, or trading a good for itself). Equating behaviorally equivalent strategies—for instance strategies that differ on what to accept in trade when an player sells her consumption good, which can never occur—reduces this to $2^{12} = 4,096$ distinct strategies. Since there are twenty types of agents, and assuming all players of the same type use the same strategies, there are $4096^{20} \approx 10^{72}$ pure strategies for the economy, an astronomically large figure. However simulations with 1000 to 2000 players—50 to 100 agents of each of the twenty types—exhibit quite rapid convergence. Repeated runs of the model with different random draws of the population exhibit converges to roughly the same equilibrium after about 6,000 generations, which is about 60,000 trading sessions.

A simulation begins by creating a population of agents with randomly generated genomes, empty inventories and zero wealth. In each round

of trade, each player chooses a good from inventory to sell, and trading partners are assigned randomly. If they agree to trade, each player's wealth is augmented by u_c if the player acquires its consumption good, and decremented by k_p if it must produce its production good. Whether or not trade takes place, each player's wealth is debited an amount equal to the storage costs on the good it holds in inventory.

Every ten rounds, which we call a 'generation,' some agents die and others are born. The lowest-scorers of each type dies, and are replaced by the offspring of the highest-scoring players of that type, and the offspring is endowed with the average wealth for that type. In addition, elderly players—defined as those who have lived for more than ten generations—die at a given rate and are replaced by offspring of the more successful agents. Reproduction takes the form of asexual replication with mutation (one entry in the trading matrix is randomly chosen to mutate for each new player).

A typical simulation used 100 agents of each of the twenty types, with storage costs for the five goods $(g_1, g_2, g_3, g_4, g_5)$ set to $(40, 30, 20, 10, 2)$. The simulation ran for 553 generations before all player types stabilized, in the sense that the frequencies of the various strategies remained constant over time except for the random perturbations induced by mutation. As expected, *the low-storage-cost good g_5 emerged as a universal medium of exchange.*

It may be suspected that the simulation is simply picking out agents who are partial to goods with low storage cost. Two additional simulations dispel this suspicion. In the first, we increase the frequency of g_4 -consumers from 20% to 44% of the total population and reduce the frequency of g_5 -consumers to 4% of the population. Since the probability of finding a buyer for g_4 is now significantly higher than for any other good, it is possible that g_4 serve as money for other agents, despite a storage cost that is higher than that of g_5 (for this simulation, the storage cost of g_4 was set to 3 and that of g_5 remained at 2). Using 2000 agents, the strategies stabilized after 935 generations, all agents accept g_4 as money, except g_3 -producers who consume g_1 , and only g_1 -producers and/or consumers accept g_5 as money. This pattern of trade has an intuitive interpretation: storage costs are so high for g_1 -producers that they will accept g_5 in trade, despite its drawbacks; g_1 -consumers therefore accept g_5 because they can trade it with g_1 -producers (that g_3 -producers who consume g_1 do not accept g_4 is an anomaly that did not appear in other runs). This shows that *both storage costs and frequency of use enter into the specification of media of exchange.*

In our second simulation, we increase the frequency of g_5 -producers from 20% to 68% of the total population, and we reduce the frequency of all other producer types equally, to 8% of the population each. There is now a glut of the low-storage-cost good g_5 , so we would expect agents to reject g_5 in trade, and move towards using g_4 as a money good. Again for this simulation, the storage cost of g_4 was set to 3 and that of g_5 remained at 2. Using 4000 agents, the strategies stabilized after 760 generations, all agents accept g_4 as money, and only g_1 -producers and some g_1 -consumers accept g_5 as money. As in the previous simulation, storage costs are sufficiently high for g_1 -producers that they will accept g_5 in trade despite its oversupply, and some g_1 -consumers accept g_5 because they can trade it with g_1 -producers. This shows that *in addition to storage costs the frequency of production affects whether a good is accepted as money*: a good must be scarce as well as low cost to serve as a medium of exchange.

What about *fiat* money? In this case intuition (and perhaps some familiarity with monetary history) suggests that using *fiat* money involves a *self-fulfilling prophecy*, in the sense that if enough of the population expects a *fiat* good to be accepted as money, then everyone will accept it. Our simulations show that this is indeed the case. If m is a *fiat* good, and if the initial fraction of players willing to accept m in trade is sufficiently large, then in equilibrium m is *fiat* money. In one simulation we set the amount of *fiat* money per player to one unit for every five players, and we set the player genotypes so that initially 50% of players accept m in exchange. Then in equilibrium we find that m serves as *fiat* money. Other simulations give the same result when the fiat good/player ratio is a low and one unit per ten players and as high as six units per ten agents, and when as little as 25% of the initial stock of agents are genetically constrained, at least in the initial period, to accept m in exchange. Under random assignment of genomes to players, however, the *fiat* good does not generally emerge as *fiat* money.

10.3 Good Vibrations

Consider the pure coordination game at the right. We can check using the techniques of Chapters 3 that there are two pure strategy equilibria, ll and rr , as well as a mixed strategy equilibrium. If we represent the out-of-equilibrium dynamics of the game using a replicator process (see Chapter 9), the pure strategy equilibria will be stable and the mixed strategy equilibrium unstable. But the concept of stability used, while at first glance compelling and intuitive, is actually quite unrealistic. The idea is that if we start at the equilibrium ll and we subject the system to a small disequilibrium shock, the system will move back into equilibrium. But in the real world dynamical systems are *constantly* subject to shocks, and if the shocks come frequently enough, the system won't have time to move back into equilibrium before the next shock comes.

	l	r
l	5,5	0,0
r	0,0	3,3

The evolutionary models considered in Chapters 7 and 9 are certainly subject to continual random 'shocks,' since agents are paired randomly, play mixed strategies with stochastic outcomes, and update their strategies by sampling the population. We avoided considering the stochastic nature of these processes by implicitly assuming that random variables can be replaced by their expected values, and mutations occur infrequently compared with the time to restore equilibrium. But what if we are not sure that these approximations are reasonable, or even worse, what if we know they are not?

One way to go is to move from deterministic to stochastic differential equations, where we add an appropriate stochastic error term to the right hand side of an equation like (8.1). This approach turns out to be very fruitful, but involves sophisticated mathematical techniques, including stochastic processes and partial differential equations.³ Moreover, applications have been confined mainly to financial economics. Applying the approach to game theory is very difficult, since stochastic differential equations with more than one independent variable virtually never has a closed form solution—so it's computer simulation all the way! We suggest here a completely different, and much more accessible approach, based on the work of H. Peyton Young (1998) and others. We start by modeling adaptive learning with and without errors.

³For relatively accessible expositions, see Dixit 1993 and Karlin and Taylor 1981.

10.4 Adaptive Learning

How does an agent decide what strategy to follow in a game? We have described three distinct methods so far in our study of game theory. The first is to determine the expected behavior of the other players and choose a best response (“rational expectations”). Second is to inherit a strategy (e.g., from one’s parents) and blindly play it. The third is to mimic another player by switching to the other player’s strategy, if it seems to be doing better than one’s own. But there is a fourth, and very commonly followed, *modus operandi*: follow the history of how other players have played in the past, and choose a strategy for the future that is a best response to the past play of others. We call this *adaptive learning*, or *adaptive expectations*.

To formalize this, consider an evolutionary game in which each player has limited memory, remembering only the last m moves $h = \{h_1, h_2, \dots, h_m\}$ of the players with whom he has been paired. If the player chooses the next move as a best response to h , we say the player follows adaptive learning.

Suppose, for instance, two agents play the coordination game in §10.2, but the payoffs to ll and rr are both 5, 5. Let $m = 2$, so the players look at the last two actions chosen by their opponents. The best response to ll is thus l , the best response to rr is r , and the best response to rl or lr is any combination of l and r . We take this combination to be: play l with probability $1/2$ and r with probability $1/2$. There are clearly sixteen distinct ‘states’ of the game, which we label by $abcd$, where each of the letters can be l or r , b is the last move by player 1, a is player 1’s move previous to this, d is the last move by player 2, and c is player 2’s move previous to this. For instance $llrl$ means player 1 moved l on the previous two rounds, while player 2 moved first r and then l .

We can reduce the number of states to ten by recognizing that since we don’t care about the order in which the players are counted, a state $abcd$ and a state $cdab$ are equivalent. Eliminating redundant states, and ordering the remaining states alphabetically, the states become $llll$, $lllr$, $llrl$, $llrr$, $lrlr$, $lrrl$, $lrrr$, $rlrl$, $rlrr$, and $rrrr$. Given any state, we can now compute the probability of a transition to any other state on the next play of the game. For instance, $llll$ (and similarly $rrrr$) is an *absorbing* state in the sense that, once it is entered, it stays there forever. The state $lllr$ goes to states $llrl$ and $lrlr$, each with probability $1/2$. The state $llrl$ goes either to $llll$ where it stays forever, or to $lllr$, each with probability $1/2$. The state $lrlr$ goes to $rlrl$ and $rrrr$ each with probability $1/4$, and

to $rlrr$ with probability $1/2$. And so on.

We can summarize the transitions from state to state a 10×10 matrix $M = (m_{ij})$, where $m_{abcd,efgi}$ is the probability of moving from state $abcd$ to state $efgi$. We call M a *probability transition matrix*, and the dynamic process of moving from state to state is a *Markov chain* (§16.19). Since matrices are easier to describe and manipulate if their rows and columns are numbered, we will assign numbers to the various states, as follows: $llll = 1, llrl = 2, \dots, rrrr = 10$. This gives us the following probability transition matrix:

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 \\ 0 & 0 & 0.25 & 0.25 & 0 & 0.25 & 0.25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\ 0.25 & 0.5 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Also, if we represent the ten states by the ten ten-dimensional row vectors $\{v_1, \dots, v_{10}\}$, where $v_1 = (1, 0, \dots, 0)$, $v_2 = (0, 1, 0, \dots, 0)$, and so on, then it is easy to see that if we are in state v_i in one period, the probability distribution of states in the next period is just $v_i M$, meaning the product of v_i , which is a 1×10 row vector, and M , which is a 10×10 matrix, so the product is another 1×10 row vector. It is also easy to see that the sum of the entries in $v_i M$ is unity and each entry represents the probability that the corresponding state will be entered in the next period.

If the system starts in state i at $t = 0$, $v_i M$ is the probability distribution of the state it's in at $t = 1$. Can we calculate the probability distribution of the state the system is in at $t = 2$? Well, let's write

$$v_i M = p_1 v_1 + \dots + p_{10} v_{10}.$$

Then with probability p_j the system has probability distribution $v_j M$ in the second period, so the probability distribution of states in the second period is

$$p_1 v_1 M + \dots + p_{10} v_{10} M = v_i M^2.$$

Similar reasoning shows that the probability distribution of states after k periods is simply $v_i M^k$. Thus just as M is the probability transition

matrix for one period, so is M^k the probability transition matrix for k periods. To find out the long-run behavior of the system we therefore want to calculate

$$M^* = \lim_{k \rightarrow \infty} M^k.$$

I trust you understand that this cannot be calculated by hand! Even calculating M^2 by hand is a royal drag. However there are ways of computing M^* without actually calculating the limit (see any book on Markov chains). I let Mathematica, the computer algebra software package, calculate M^k for larger and larger k until the entries in the matrix stopped changing or became vanishingly small, and this is what I came up with:

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2/3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/3 \\ 5/6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/6 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2/3 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 \\ 1/6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5/6 \\ 2/3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/3 \\ 1/3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2/3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

In other words, no matter where you start, you end up in one of the absorbing states, which is a Pareto-optimal Nash equilibrium. We call pure strategy Nash equilibria in which all players choose the same strategy *conventions* (Young 1998). We conclude that *adaptive learning leads with probability one to a convention*.

10.4.1 The Steady State of a Markov Chain

There is a simpler way to compute M^* in the previous case. The computation also gives a better intuitive feel for the steady state solution to the adaptive learning dynamical system generated by a pure coordination game. We know whatever state we start the system in, we will end up in either state $llll$ or state $rrrr$. For state $abcd$, let $P[abcd]$ be the probability that we end up in lll starting from $abcd$. Clearly $P[llll] = 1$ and $P[rrrr] = 0$. Moreover, $P[lllr] = P[llrl]/2 + P[lrrl]/2$, since $lllr$ moves to either $llrl$ or to $lrrl$ with equal probability. Generalizing, you can check that if we define

$$v = (P[llll], P[lllr], \dots, P[rrrr])',$$

the column vector of probabilities of being absorbed in state lll , then we have

$$Mv = v.$$

If we solve this equation for v , subject to $v[1] = 1$, we get

$$v = (1, 2/3, 5/6, 1/2, 1/3, 1/2, 1/6, 2/3, 1/3, 0)',$$

which then must be the first column of M^* . The rest of the columns are zero, except for the last which must have entries so the rows each sum up to unity. By the way, I wouldn't try to solve the equation $Mv = v$ by hand unless you're a masochist. I let Mathematica do it (v is called an *eigenvector* of M , so Mathematica has a special routine for finding v easily).

10.5 Adaptive Learning When not all Conventions are Equal

Now consider the pure coordination game illustrated in §10.2, where the ll convention Pareto-dominates the rr convention. How does adaptive learning work in such an environment? We again assume each player finds a best response to the history of the other player's previous two moves. The best response to ll and rr are still l and r , respectively, but now the best response to rl or lr is also l . Now, for instance llr and lr both go to lll with probability 1. The probability transition matrix now becomes

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

To calculate

$$M^* = \lim_{k \rightarrow \infty} M^k,$$

is relatively simple, since in this case $M^k = M^4$ for $k \geq 4$. Thus we have

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

In other words, if you start in state $rrrr$ you stay there; otherwise after four steps, you arrive at $llll$ and remain there forever. We conclude that *with adaptive learning, if the system starts in a non-conventional state, it always ends up in the Pareto-efficient conventional state.*

10.6 Adaptive Learning in a Pure Coordination Game with Errors

We now investigate the effect on a dynamic adaptive learning system when players are subject to error. Consider the pure coordination game illustrated in §10.2, but where the payoffs to ll and rr are equal. Suppose each player finds a best response to the history of the other player's previous two moves with probability $1 - \epsilon$, but chooses incorrectly with probability $\epsilon > 0$. The probability transition matrix now becomes

$$M = \begin{pmatrix} a & 2b & 0 & 0 & e & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c & d & 0 & c & d & 0 & 0 & 0 \\ c & 1/2 & 0 & 0 & d & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b & e & 0 & a & b & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a & 0 & 1/4 & 1/2 & 1/4 \\ 0 & 0 & 1/4 & 1/4 & 0 & 1/4 & 1/4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & d & 1/2 & c \\ 1/4 & 1/2 & 0 & 0 & 1/4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & d & d & 0 & c & c & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & e & 2b & a \end{pmatrix}$$

where $a = (1 - \epsilon)^2$, $b = \epsilon(1 - \epsilon)$, $c = (1 - \epsilon)/2$, $d = \epsilon/2$, and $e = \epsilon^2$. Notice that now *there are no absorbing states*. To see what happens in the long run, suppose $\epsilon = 0.01$, so errors occur one percent of the time.

Using Mathematica to calculate M^* , we find *all the rows are the same*, and each row has the entries

$$(0.4420.0180.0180.0010.00020.0350.0180.00020.0180.442)$$

In other words, you spend about 88.4% of the time in one of the conventional states, and about 11.6% of the time in the other states.

It should be intuitively obvious how the system behaves. If the system is in a conventional state, say lll , it remains there in the next period with probability $(1 - \epsilon)^2 = 98\%$. If one player makes an error, the state moves to llr . If there are no more errors for a while, we know it will return to lll eventually. Thus it requires multiple errors to ‘kick’ the system to a new convention. For instance, $lll \rightarrow llr \rightarrow lrrr \rightarrow rrrr$ can occur with just two errors: $lll \rightarrow llr$ with one error, $llr \rightarrow lrrr$ with one error, and $lrrr \rightarrow rrrr$ with no errors, but probability $1/2$. We thus expect convention flips about every 200 plays of the game.

To test our ‘informed intuition,’ I simulated 1000 repetitions of this stochastic dynamical system using Mathematica. Figure 10.1 reports on the result.

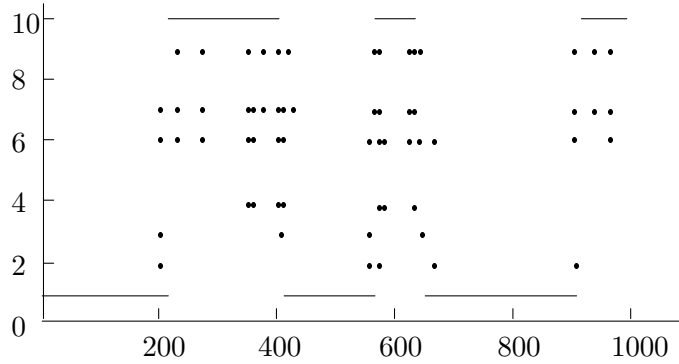


Figure 10.1: Simulating a Stochastic Dynamical System I

10.7 Stochastic Stability

We define a state in a stochastic dynamical system to be *stochastically stable* if the long-run probability of being in that state does not become zero or vanishingly small as the rate of error ϵ goes to zero. Clearly in the previous example lll and $rrrr$ are both stochastically stable, and no other state is. Consider the game in 10.2. It would be nice if the Pareto-dominant equilibrium ll were stochastically stable, and no other

state were stochastically stable. We shall see that is the case. Now the probability transition matrix becomes

$$M = \begin{pmatrix} a & 2b & 0 & 0 & e & 0 & 0 & 0 & 0 & 0 \\ 0 & 2b & a & 0 & e & 0 & 0 & 0 & 0 & 0 \\ a & 2b & 0 & 0 & e & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b & e & 0 & a & b & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & b & a & b \\ 0 & 0 & a & b & 0 & b & e & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & b & a & b \\ a & 2b & 0 & 0 & e & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b & e & 0 & a & b & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & e & 2b & a \end{pmatrix}$$

where $a = (1-\epsilon)^2$, $b = \epsilon(1-\epsilon)$, and $e = \epsilon^2$. Again there are no absorbing states. If $\epsilon = 0.01$, we calculate M^* , again finding that we find *all the rows are the same*, and each row has the entries

$$(0.9605 \quad 0.0198 \quad 0.0198 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0)$$

In other words, the system spends 96% of the time in the Pareto-dominant conventional states, and virtually all of the remaining time in ‘nearby states.’ It is clear (though it should be formally proved) that ll is the only stochastically stable state.